Amendments to the CLAIMS:

Claims 1-27: Previously Cancelled
Claims 31-36: Cancelled
Allowed Claims: 28-30 & 37
Currenly Post Allowance Amended Claims: 28 & 37

The embodiments of the invention in which an exclusive property or privilege is claimed are defined as follows:

28. (Currently Amended) A method of forming/defining and solving a loadflow computation model of a power network to affect control of voltages and power flows in a power system, comprising the steps of:

obtaining on-line/simulated data of open/close status of all switches and circuit breakers in the power network, and reading data of operating limits of components of the power network including maximum power carrying capability limits of transmission lines, transformers, and PV-node, a generator-node where Real-Power-P and Voltage-Magnitude-V are given/assigned/specified/set, maximum and minimum reactive power generation capability limits of generators, and transformers tap position limits,

obtaining on-line readings of given/assigned/specified/set Real-Power-P and Reactive-Power-Q at PQ-nodes, Real-Power-P and voltage-magnitude-V at PV-nodes, voltage magnitude and angle at a reference/slack node, and transformer turns ratios, wherein said on-line readings are the controlled variables/parameters,

initiating loadflow computation with initial approximate/guess solution of the same voltage magnitude and angle as those of the reference/slack node for all the PQ-nodes and the PV-nodes, and said initial approximate/guess solution is referred to as a slack-start,

performing loadflow computation to calculate, depending on loadflow computation model used, complex voltages or their real and imaginary components or voltage magnitude corrections and voltage angle corrections at nodes of the power network providing for calculation of power flow through different components of the power network, and to calculate reactive power generation and transformer tap-position indications,

decomposing the power network for performing said loadflow computation in parallel by a method referred to as Suresh's diakoptics that involves determining a sub-

power-network for each node involving directly connected nodes referred to as level-1 nodes and directly connected nodes to level-1 nodes referred to as level-2 nodes, and a level of outward connectivity for local solution of a sub-power-network around a given node is determined experimentally,

initializing, at the beginning of each new iteration, a vector of dimension equal to the number of nodes in the power network with each element value zero,

solving all sub-networks in parallel using available solution estimate at the start of ~~the~~ each iteration,

adding newly calculated solution estimates or corrections to the available solution estimate for a node resulting from different sub-networks, 'q' number of sub-networks, in which a node is contained, in a corresponding vector element that gets initialized zero at the beginning of each new iteration,

counting the number of additions and calculating new solution estimate or corrections to the available solution estimate by taking the average or root mean square value using any relevant relations (30) to (39) in the following depending on the loadflow computation method used, and

storing the new solution estimate at the end of the current iteration as ~~initial~~ available solution estimate at the start of ~~for~~ the next iteration,

wherein the said Suresh's diakoptics method uses the following relations,

$$V_p^{(r+1)} = (V_{p1}^{(r+1)} + V_{p2}^{(r+1)} + V_{p3}^{(r+1)} + \ldots + V_{pq}^{(r+1)})/q \tag{30}$$

$$\Delta\theta_p^{(r+1)} = (\Delta\theta_{p1}^{(r+1)} + \Delta\theta_{p2}^{(r+1)} + \Delta\theta_{p3}^{(r+1)} + \ldots + \Delta\theta_{pq}^{(r+1)})/q \tag{31}$$

$$\Delta V_p^{(r+1)} = (\Delta V_{p1}^{(r+1)} + \Delta V_{p2}^{(r+1)} + \Delta V_{p3}^{(r+1)} + \ldots + \Delta V_{pq}^{(r+1)})/q \tag{32}$$

$$e_p^{(r+1)} = (e_{p1}^{(r+1)} + e_{p2}^{(r+1)} + e_{p3}^{(r+1)} + \ldots + e_{pq}^{(r+1)})/q \tag{33}$$

$$f_p^{(r+1)} = (f_{p1}^{(r+1)} + f_{p2}^{(r+1)} + f_{p3}^{(r+1)} + \ldots + f_{pq}^{(r+1)})/q \tag{34}$$

wherein relations (30) to (34), can also alternatively be written as relations (35) to (39) as below,

$$\mathbf{V}_p^{(r+1)} = \sqrt{(\mathrm{Re}((\mathbf{V}_{p1}^{(r+1)})^2) + \mathrm{Re}((\mathbf{V}_{p2}^{(r+1)})^2) + \ldots + \mathrm{Re}((\mathbf{V}_{pq}^{(r+1)})^2)/q}$$

$$+ j\sqrt{(\mathrm{Im}((\mathbf{V}_{p1}^{(r+1)})^2) + \mathrm{Im}((\mathbf{V}_{p2}^{(r+1)})^2) + \ldots + \mathrm{Im}((\mathbf{V}_{pq}^{(r+1)})^2))/q} \qquad (35)$$

$$\Delta\theta_p^{(r+1)} = \sqrt{(\Delta\theta_{p1}^{(r+1)})^2 + (\Delta\theta_{p2}^{(r+1)})^2 + \ldots + (\Delta\theta_{pq}^{(r+1)})^2)/q} \qquad (36)$$

$$\Delta V_p^{(r+1)} = \sqrt{(\Delta V_{p1}^{(r+1)})^2 + (\Delta V_{p2}^{(r+1)})^2 + \ldots + (\Delta V_{pq}^{(r+1)})^2)/q} \qquad (37)$$

$$e_p^{(r+1)} = \sqrt{((e_{p1}^{(r+1)})^2 + (e_{p2}^{(r+1)})^2 + \ldots + (e_{pq}^{(r+1)})^2)/q} \qquad (38)$$

$$f_p^{(r+1)} = \sqrt{((f_{p1}^{(r+1)})^2 + (f_{p2}^{(r+1)})^2 + \ldots + (f_{pq}^{(r+1)})^2)/q} \qquad (39)$$

wherein, square of any positive or negative number being positive, if the original not-squared value of any number is negative, the same algebraic sign is attached after squaring that number, and if the mean of squared values turns out to be a negative number, negative sign is attached after taking the square root of the unsigned number, V_p, θ_p are voltage magnitude and voltage angle at node-p, e_p and f_p are the real and imaginary parts of the complex voltage $\mathbf{V}_p$ of node-p, words "Re" means "real part of" and words "Im" means "imaginary part of", symbol Δ before any of defined electrical quantities defines the change in the value of electrical quantity, and superscript 'r' indicates the iteration count,

evaluating loadflow computation for any over loaded components of the power network and for under/over voltage at any of the nodes of the power network,

correcting one or more controlled variables/parameters and repeating the performing loadflow computation by decomposing, initializing, solving, adding, counting, storing, evaluating, and correcting steps until evaluating step finds no over loaded components and no under/over voltages in the power network, and

affecting a change in power flow through components the power network and voltage magnitudes and angles at the nodes of the power network by actually implementing the finally obtained values of controlled variables/parameters after evaluating step finds a good power system or stated alternatively the power

network without any overloaded components and under/over voltages, which finally obtained controlled variables/parameters however are stored for acting upon fast in case a simulated event actually occurs.

29. A method as defined in claim-1 wherein the loadflow computation method referred to as Gauss-Seidel-Patel Loadflow (GSPL) computation method is characterized in using self-iteration denoted by 'sr' within a network-wide/sub-network-wide global iteration depicted by 'r' in the GSPL model defined by equation (27) given in the following,

$$(V_p^{(sr+1)})^{(r+1)} = [\{(PSH_p - jQSH_p)/((V_p^*)^{sr})^r\} - \sum_{q=1}^{p-1} Y_{pq}V_q^{(r+1)} - \sum_{q=p+1}^{n} Y_{pq}V_q^r] / Y_{pp} \qquad (27)$$

wherein, PSH_p and QSH_p are scheduled/specified/known/set real and reactive power, V_p is the complex node-p voltage, and Y_{pq} and Y_{pp} are off-diagonal and diagonal complex elements of the network admittance matrix.

30. A method as defined in claim-28 wherein a parallel loadflow computation is performed using a parallel computer: a server processor-array processors architecture, wherein each of the array processors send communication to and receive communication from only the server processor, commonly shared memory locations, and each processor's private memory locations, but not among themselves.

31. CANCELLED

32. CANCELLED

33. CANCELLED

34. CANCELLED

35. CANCELLED

36. CANCELLED

37. (Currently Amended) A multiprocessor computing apparatus for performing the said parallel loadflow computation as defined in claim-28 comprising in combination:

a plurality of processing units adapted to receive and process data, instructions and control signals, and connected to common system bus in parallel asynchronous fashion;

a plurality of local private main memory means for storing data, instructions and control signals, each said <u>local private</u> main memory means being directly and

asynchronously connected to each said processing unit;

common shared memory coupled directly to said common system bus for sending/receiving data, instructions and control signals asynchronously to/from each said processing unit, without providing inter-processor communications;

I/O adapter/control unit coupled directly to a main/server processor which is one of the said plurality of processing units;

wherein I/O adapter/control units in plurality and different one of which is coupled directly to each of the said plurality of processing units physically located at far distances in case of said multiprocessor computing apparatus is organized for distributed processing.